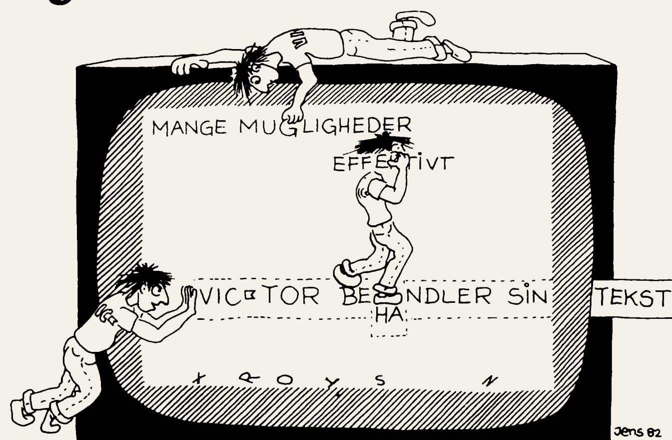
VI & VIC

Nr. 5 DECEMBER 1982 - JANUAR 1983 Pris 10,95 kr.

Tekstbehandling på VIC kan spare mange timers arbejde

For den VIC-bruger, der er i besiddelse af en printer og diskettestation, er det ingen sag at komme i gang med tekstbehandling.
Commodore Data vil inden for meget kort tid præsentere et nyt tekstbehandlingsprogram til VIC. Det er ikke et program, man kan sammenligne med de meget omfattende tekstbehandlingsprogrammer, der leveres til store datasystemer. Men til gengæld er prisen uhørt lav: under 700 kr. incl. moms.
Sammen med programmet får man den originale, engelske brugervejledning - men der er også udarbejdet et dansk koncentrat af denne vejledning, så der skulle ikke være nogen problemer i forbindelse med brugen af programmet.



Mange muligheder

I begrebet tekstbehandling ligger der faktisk kun det, at man med sådanne systemer er i stand til at skrive en tekst ind i computerens hukommelse - og, hvad der nok er væsentligt: Bagefter kan man frit ændre i teksten, man kan flytte dele af teksten fra et sted til et andet, og man kan i det hele taget undgå at gøre det samme arbejde mere end én gang.
I forbindelse med redigering af tekster er det en stor arbejdslettelse at benytte tekstbehandlingsprogrammet. Når man har printet teksten ud på papir, kan man i ro og mag læse den igennem, for at finde eventuelle fejl. Herefter er det ikke noget stort problem at foretage de nødvendige rettelser på skærmen, hvorefter man på ny kan printe teksten ud på papir - denne gang i den korrekte form.

Genbrug

Den helt store fordel med tekstbehandlingsprogrammet er, som allerede nævnt, at man ikke behøver at udføre det samme arbejde mere end én gang. Skal man f.eks. skrive enslydende breve til flere personer, kan man nøjes med at skrive brevet én gang. Når man har printet det ud på papir, kan man nøjes med at rette navn og adresse, inden man printer den samme tekst ud en gang til.
En anden anvendelsesmulighed for tekstbehandlingsprogrammet ligger i at oprette et adressekartotek, så man automatisk kan skrive et større eller mindre antal adresser ud på selvklæbende etiketter.

Fortsættes side 2

Arbejde med relative filer

Læs side 4-5

Lav dine egne tegn til VIC

Læs side 7

Specialsider om maskinsprog



Læs fra side 11

Fortsat fra forsiden

Det kan programmet

Teksbehandlingsprogrammet til VIC kan stort set udføre alle de vigtigste funktioner for et sådant program. Hvor megen tekst, der kan arbejdes med på en gang, afhænger af, hvor meget hukommelse, man har til rådighed. Teksten bygges op på en enkelt »side«, men denne side kan til gengæld være meget stor. Brugeren af VIC tekstbehandlingsprogrammet fastsætter selv sidens bredde fra 20-80 tegn pr. linie. Den hukommelse, der er koblet til computeren, sætter grænsen for, hvor mange linier, der kan blive plads til. Da redaktionen testede programmet, havde vi sluttet ekstra 16K til VIC-20, og der var ingen vanskeligheder med at arbejde med tekster over 300 linier af 60 tegn.

Det tog to arbejdsdage at finde den vindende løsning på »terningen«

Der var virkelig mange løsninger på VI & VIC's konkurrence om et program, der kan gøre det ud for en »professor-terning«, og det viste sig meget hurtigt, at problemet med at kreere et passende skærmbillede virkelig var én af de vanskeligheder, der først måtte overvindes, inden man kunne tage fat på at løse det egentlige problem: At dreje de seks sider.

I bedømmelsen af de indsendte løsninger blev det derfor meget hurtigt de to ting, der først tiltrak sig opmærksomheden: Den grafiske fremstilling på skærmen og betjeningen af funktionerne.



John Nielsen, Rødby: - Det tog mig to hele dage at få programmet færdigt. (Foto: Anders Knudsen).

Udfaldet

Ganske mange valgte at lade terningen fremtræde i »udfoldet« stand, hvilket giver et godt overblik over hele terningen - men som til gengæld kan være vanskelig at kigge på, når man skal beslutte sig for, hvilken side man skal dreje. Den gruppe løsninger, der var bygget op omkring en terning i perspektiv, så man kan se de tre sider på én gang har den svaghed, at man ikke kan overskue hele terningen på én gang, men det har vo konkurrences vinder løst ved at indbygge en funktion i programmet, som vender hele terningen, så man i stedet ser de tre sider, som før var skjult.

Vi bringer programudlistningen for den vindende terning her på siden sammen med vinderen, John Nielsens oplysning om, at den fylder 2202 bytes.

To dage

John Nielsen, der bor i Rødby, fortæller iøvrigt til VI & VIC, at det tog ham to dage at løse opgaven - men efter, at

Program, der LØSER en blandet terning

Den opgave, VI & VIC i sin tid stillede sine læsere gik alene ud på at simulere professor-terningens funktion, for vi var beskedne nok til at holde os fra opgaven at løse en terning.

Men der er andre, der har kastet sig over dette problem.

Det amerikanske tidsskrift Compute har således i dette efterår offentliggjort et program, der kan klare en blandet terning, men at opgaven virkelig er stor, fremgår af, at programmet fylder 14K - og udlistningen fylder fem hele sider i VI & VIC-størrelse.

VI & VIC

VI & VIC udgives af commodore data a/s,
Bjerrevej 67, 8700 Horsens.
Redaktion og tilrettelæggelse:
Baghuset Grafik/Design, Vejle.
Ansvarshavende redaktør: Steen Venbjerg.
Tryk: Baghuset Grafik/Design, Vejle.
Giro nr. 6 17 39 93

ISSN 0108-2213

han havde postet løsningen, opdagede han en fejl ved programmet.
For at få den rettet måtte han tage opstilling ved postkassen, hvor han så overtalte kasse-tømmeren til at give ham brevet tilbage, så fejlen kunne blive rettet.
Om sin baggrund for at programmere siger John Nielsen, at han har lært lidt på navigationsskolen, men at han iøvrigt føler sig som en novice på VIC. Han anskaffede sin computer i februar, men han er alligevel kommet så langt, at han var begyndt at spekulere i maskinkode. Så præmien, maskinkodemonitoren VICMON kommer meget tilpas.

God grafik

En af løsningerne på professor-terningen indeholdt en lille oversigtsmodel, der viste alle seks sider på én gang - og det bemærkelsesværdige var, at de firkanter, terningen var bygget op af, var en smule mindre end den velkendte cursor eller et negativt mellemrum (POKE 160).
Denne effekt havde løseren af opgaven opnået ved at bruge en af de fire hjørnekarakterer (POKE 76, 79, 80 eller 122) i den negative udgave, som man finder frem til ved at lægge 128 til POKE-værdien. Altså i dette tilfælde 204, 207, 208 eller 250.

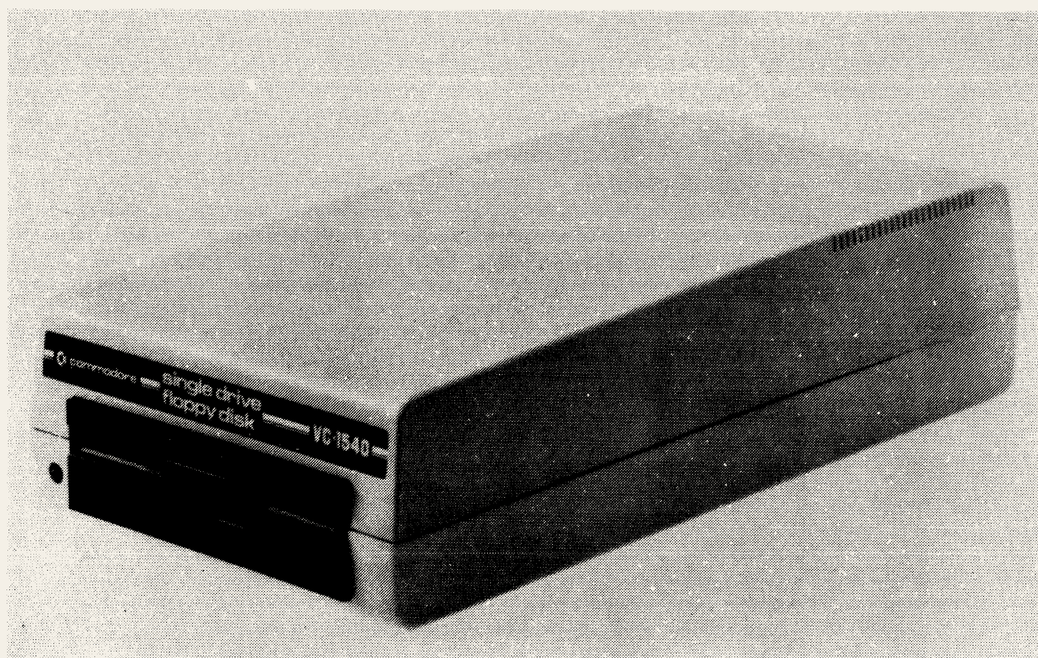
```
0 REM**PROFESSOR TERNING**
2 A1=1:A2=2:A3=3:PRINT"▯":POKE36879,238:DIMT(3,3,6):D=7860:E=38556
4 FORX=1TO6:FORX1=1TO3:FORX2=1TO3:T(X2,X1,X)=X3:NEXTX2,X1
6 READX3:NEXTX:F=38740:G=38439:H=38591
8 IFPEEK(4096)<>0THEND=4276:E=38044:F=38228:G=37927:H=38079
10 FORY=22TO110STEP44:FORX=0TO2:POKED+X*2+Y,160:NEXTX,Y
12 FORY=24TO108STEP42:FORX=2TO6STEP2:POKED+X-Y,233:POKED+X+1-Y,105:NEXTX,Y
14 FORY=1TO3:FORX=2TO6STEP2:POKED+3+X*22-Y*42,233:POKED+25+X*22-Y*42,105:NEXTX,Y
16 PRINT"███PROFESSOR-TERNING█"
18 IFV=0THEN30
20 PRINT"██████████3███2███1███3█2█1█5█5█4":A1=6:A2=5:A3=4:V=1
22 FORY=1TO3:FORX=1TO3:POKEF-2*X-44*Y,T(X,Y,A1):NEXTX,Y
24 FORY=1TO3:FORX=2TO6STEP2:POKEG-X+42*Y,T(X/2,Y,A2):POKEG-X-1+42*Y,T(X/2,Y,A2):
NEXTX,Y
26 FORY=1TO3:FORX=2TO6STEP2:POKEH+22-X*22+Y*42,T(X/2,Y,A3):POKEH-X*22+Y*42,T(X/2
,Y,A3)
28 NEXTX,Y:GOTO40
30 PRINT"██████████1███2███3███1█2█3█4█5█6":A1=1:A2=2:A3=3:V=0
32 FORY=1TO3:FORX=1TO3:POKEE+2*X+44*Y,T(X,Y,A1):NEXTX,Y
34 FORY=1TO3:FORX=2TO6STEP2:POKEE+42+X-42*Y,T(X/2,Y,A2):POKEE+42+X+1-42*Y,T(X/2,
Y,A2):NEXTX,Y
36 FORY=1TO3:FORX=2TO6STEP2:POKEE+27+X*22-Y*42,T(X/2,Y,A3):POKEE+49+X*22-Y*42,T(
X/2,Y,A3)
38 NEXTX,Y
40 PRINT"████████████████████O█P,█N█ED,█V█ENSTRE,█H█OJRE + NUMMER EL. █D█REJ  ?█
█ ";
42 GETA$:IFA$=""OR(A$<>"O"ANDA$<>"N"ANDA$<>"V"ANDA$<>"H"ANDA$<>"D")THEN42
44 IFA$="D"ANDV=0THEN20
46 IFA$="D"ANDV=1THEN30
48 PRINTA$;
50 GETB$:IFB$=""THEN50
52 IFASC(B$)=20THENPRINTB$;:GOTO42
54 IFASC(B$)<49ORASC(B$)>54THEN50
56 B=VAL(B$):IFA$="O"ANDB<4THENI=5:J=6:K=2:L=1:GOTO66
58 IFA$="V"THENI=3:J=6:K=4:L=1:M=3:N=1:O=1:P=2:R=2:Q=3:Z=1:GOTO92
60 IFA$="H"THENI=3:J=6:K=4:L=1:M=1:N=3:O=2:P=1:R=3:Q=2:Z=3:GOTO92
62 IFB>3THENB=B-3:GOTO76
64 I=1:J=2:K=6:L=5
66 FORX=1TO3:A4=T(B,X,I):T(B,X,I)=T(B,4-X,J):T(B,4-X,J)=T(B,4-X,K)
68 T(B,4-X,K)=T(B,X,L):T(B,X,L)=A4:NEXT:IFB=2THEN18
70 C=3:IFB=1THENC=4
72 IFA$="O"THENM=3:N=1:O=1:P=2:R=2:Q=3:GOTO86
74 M=1:N=3:O=2:P=1:R=3:Q=2:GOTO86
76 IFA$="O"THENI=3:J=5:K=4:L=2:M=3:N=1:O=1:P=2:R=2:Q=3:GOTO80
78 I=2:J=4:K=5:L=3:M=1:N=3:O=2:P=1:R=3:Q=2
80 FORX=1TO3:A4=T(X,B,I):T(X,B,I)=T(4-X,B,J):T(4-X,B,J)=T(4-X,B,K):T(4-X,B,K)=T(
X,B,L)
82 T(X,B,L)=A4:NEXT:IFB=2THEN18
84 C=1:IFB=3THENC=6
86 A(1)=T(1,1,C):T(1,1,C)=T(M,N,C):T(M,N,C)=T(3,3,C):T(3,3,C)=T(N,M,C):T(N,M,C)=
A(1)
88 A(1)=T(O,P,C):T(O,P,C)=T(P,O,C):T(P,O,C)=T(Q,R,C):T(Q,R,C)=T(R,Q,C):T(R,Q,C)=
A(1)
90 GOTO18
92 FORY=1TOZ:FORX=1TO3:A4=T(B,X,I):T(B,X,I)=T(4-X,B,J):T(4-X,B,J)=T(B,4-X,K)
94 T(B,4-X,K)=T(X,B,L):T(X,B,L)=A4:NEXTX,Y:IFB=2THEN18
96 C=2:IFB=3THENC=5
98 GOTO86
100 DATA1,2,5,6,4,3
```

Vinderne af VI & VIC's konkurrence

Virkelig mange løsninger både på professor-terningen og spil nåede frem til VI & VIC's redaktion. Valget var ikke let, da vinderne skulle udpeges, for der var virkelig mange interessante løsninger imellem. Men en afgørelse skulle træffes, og i terning-konkurrencen blev vinderen John Nielsen, Kirkealle 33, Rødby, som modtager maskinkodemonitoren VICMON som præmie for den professorterning-simulator, der er listet her på siden.
I Spil-afdelingen faldt præmien, en super-expander, til Arno Greve Andersen, Vibækvej 15, Odense SV. Ét af Arne Greve Andersens spil er også gengivet her i bladet.
Vi siger tillykke til vinderne og tak til alle, der havde lyst at lege med.

Diskettestationen til VIC kan også bruges til at håndtere relative filer

VIC diskettestationen kan mere, end den lover. I artiklen beskrives det, hvordan man arbejder med relative filer.



Der lægges et stort arbejde i brugervejledningerne til de forskellige tilbehørsdele til VIC. De oversættes ikke uden videre fra originalsproget, men der arbejdes også med udstyret, så man er sikker på, at den danske vejledning giver et tilstrækkeligt godt billede af de mange muligheder.

Under arbejdet med diskettestationen VC 1540 har systemkonsulent Åge Monrad således afsløret, at den faktisk kan mere end den lover. Det viser sig nemlig, at det er muligt at arbejde med relative filer og ikke udelukkende med sekventielle filer. Forskellen mellem de to filformer er, at man i en relativ kan hente en enkelt post frem, nyskrive en eller foretage rettelser uden at beskæftige sig med andre poster. I en sekventiel fil er man nødt til at læse de foregående.

Mindre programmering

Skal man f. eks. have fat i post nummer 500, må man i en sekventiel fil »traske« gennem de 499 og læse dem, inden man kommer til den post, man ønsker. Det er ikke nødvendigt når der er tale om relative filer. Den mest oplagte fordel ved de relative filer er, at de kræver langt mindre programmering. Her vises nogle program-eksempler på, hvordan man opretter, skriver, læser og udvider i relative filer.

I relative filer kan det også lade sig gøre at sortere posterne efter forskellige kriterier, og det vil man finde anvisning på i den danske brugervejledning, som er lige på trapperne.

OPRET RELATIV FIL

```
1 REM OPRETTELSE AF EN RELATIV FIL
2 REM 20 I OPEN INSTRUKTIONEN ER POSTLAENGDEN
3 REM LINIE 30. CHR$(2) ER TIL DET LOGISKE FILNUMMER SOM FINDES I OPEN INSTRUKTI
ONEN
4 REM LINIE 30. CHR$(100) ER LOW-BYTE AF POSTNUMMERET. CHR$(0) ER HIGH-BYTE AF P
OSTNR.
5 REM CHR$(1) ANGIVER HVOR I POSTEN MAN ØNSKER AT STARTE.
6 REM HELE LINIEN BRUGES TIL AT FINDE DEN ENKELTE POST, SOM SKAL LAESES/SKRIVES
FREM.
7 REM NAAR POSTEN ER FUNDET KAN MAN SAA LAESE ELLER SKRIVE MED INPUT# ELLER GET#
 ELLER
8 REM MAN KAN SKRIVE MED PRINT#.
9 REM CHR$(1) ANGIVER HVOR I POSTEN MAN ØNSKER AT STARTE.
10 OPEN 15,8,15
20 OPEN 2,8,2,"RELFIL,L,20"
25 GOSUB 130
30 PRINT#15,"P"+CHR$(2)+CHR$(100)+CHR$(0)+CHR$(1)
50 GOSUB 130
80 PRINT#2,CHR$(255);
90 GOSUB 130
100 CLOSE 2
110 CLOSE 15
120 END
125 REM ***** FEJLCHECK. SKAL BRUGES EFTER HVER DISKOPERATION. *****
130 INPUT#15,EN,EM$,ET,ES
140 IF EN<20 THEN RETURN
150 IF EN=50 THEN RETURN
```

SKRIV I RELATIV FIL

```
1 REM PROGRAMMET SKRIVER TEKSTEN 'POST NR 25' I POST 25.
10 OPEN 15,8,15
20 OPEN 2,8,2,"RELFIL,L,20"
25 GOSUB 130
30 PRINT#15,"P"+CHR$(2)+CHR$(25)+CHR$(0)+CHR$(1)
50 GOSUB 130
80 PRINT#2,"POST NR 25"
90 GOSUB 130
100 CLOSE 2
110 CLOSE 15
120 END
130 INPUT#15,EN,EM$,ET,ES
140 IF EN<20 THEN RETURN
150 IF EN=50 THEN RETURN
160 STOP

READY.

 31
```

LÆS RELATIV FIL

```
1 REM UDLAESER POST NR25 OG SKRIVER DEN PAA SKAERMEN.
10 OPEN 15,8,15
20 OPEN 2,3,2,"RELFIL"
30 GOSUB 110
40 PRINT#15,"P"+CHR$(2)+CHR$(25)+CHR$(0)+CHR$(1)
50 GOSUB 110
60 INPUT#2,A$
70 GOSUB 110
71 PRINT A$
75 CLOSE 2
80 CLOSE 15
100 END
110 INPUT#15,EN,EM$,ET,ES
120 IF EN<20 THEN RETURN
130 IF EN=50 THEN RETURN
140 STOP
```

UDVID RELATIV FIL

```
1 REM UDVIDELSE AF EN FIL TIL 200 POSTER
2 REM HIGH-BYTE = 1 (255 POSTER) DERTIL KOMMER LOW-BYTE 145.
3 REM 255+145=400
10 OPEN 15,8,15
20 OPEN 2,8,2,"FILOPRET"
30 GOSUB 110
40 PRINT#15,"P"+CHR$(2)+CHR$(145)+CHR$(1)+CHR$(1)
50 GOSUB 110
60 PRINT#2,CHR$(255);
70 GOSUB 110
80 CLOSE 2
90 CLOSE 15
100 END
110 INPUT#15,EN,EM$,ET,ES
120 IF EN<20 THEN RETURN
130 IF EN=50 THEN RETURN
```

Diskette-station VIC 2031

Der er nu kommet endnu en diskettestation til VIC 20 på markedet. Betegnelsen er VIC 2031, og i teknisk og kapacitetsmæssig henseende er den identisk med den velkendte VC 1540.
Den nye diskettestation er bedre egnet til opgaver, hvor den skal klare vedvarende belastninger. F. eks. i forbindelse med administrative opgaver, hvor diskettestationen belastes mange timer i træk.
Prisen på den nye diskettestation er 7925 kr., mens den kendte model sælges for 6495 kr.



Monitorer i sort/hvid og i farve

VIC-forhandlerne har fået en aftale med Sanoy om levering af dataskærme og farve-tv til databrug. Priserne på disse apparater ligger i intervallet fra 1899 til 5999 kr.
På samme måde kan VIC forhandlerne levere 12 og 14 tommers sort/hvide apparater.

To VIC-programmer med klar adresse til gymnasiet og HF

VIC-graf er udformet på en fast ROM-program kapsel, og det har den fordel, at programmet straks er klar til brug, når man tænder maskinen.
Programmet er udarbejdet til funktionsundersøgelse og er velegnet som pædagogisk hjælpemiddel fra folkeskolens 8. klasse niveau op til 2. gymnasie klasse. Både for 1. HF og 1. G sproglig vil programmets kurvetegning være meget fin. Ingen kan undgå at blive interesseret, når kurven bliver tegnet med tilhørende lydillustration.
Ud over kurvetegning er det muligt, at beregne nulpunkter, maksimums- og minimumsværdier, samt integral over et interval.
Det siger sig selv, at også til hjemmebrug vil VIC-graf være til stor nytte som inspirationskilde og kontrol.
På minus siden skal dog nævnes, at man kun kan tegne en graf ad gangen, det udelukker muligheder for at sammenligne to funktioner.
Vejledningen er klar og letfattelig.
VIC—stat er ligesom VIC-graf på en fast ROM-kapsel. Programmet har her den fordel, at man kan bruge maskinen på sædvanlig måde, men med VIC-stat får man udvidet basic med 15 kommandoer. Man kan få et talmateriale illustreret med stolpediagram - lodret eller vandret efter ønske - og få beregnet middelværdi og spredning. Programmet giver også mulighed for bestemmelse og bedste rette linie (liniær regression) gennem en række punkter.

VIC-stat vil være til god hjælp, når man sidder og skal skrive rapport over et fysikforsøg.

Vejledningen kunne være mere klar.

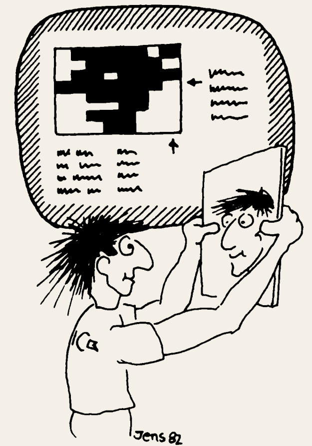
Programmers Aid og superexpander kan bruges sammen

Man har gode chancer for at løbe ind i vanskeligheder, hvis man på samme tid har tilsluttet Superexpander og programmers Aid til VIC.
De to moduler bruger begge to funktionstasterne, og her er anvisningen på, hvordan man skifter fra det ene sæt funktioner til det andet:
Fra Programmers Aid til Superexpander kommer man ved at trykke RUN/STOP og RESTORE knapperne ned på samme tid.
Fra Superexpander til Programmers Aid kommer man ved at benytte EDIT og RETURN eller PROG og RETURN.
Programmers Aid kender kun de normale BASIC-instruktioner, og det vil gå galt, hvis man søger (FIND) efter en Superexpander-instruktion, f.eks. GRAPHIC, DRAW eller PAINT), og på samme måde vil det gå galt, hvis man forsøger at bytte CHANGE) en Superexpander-instruktion.
Risikoen, man løber ved at blande de to moduler sammen, begrænser sig til at maskinen »går ned«, så man må slukke og tænde igen - men det kan jo også være ærgerligt nok.

FORTH på VIC giver helt nye muligheder for at programmere

VIC programmeres i BASIC, men sådan behøver det ikke blive ved med at være.
Et nyt modul til direkte montage i VIC eller Motherboard hedder VIC-FORTH og det giver helt nye muligheder for programmering.
VIC-FORTH er et kraftigt operativsystem og programmeringssprog, som på mange måder adskiller sig væsentligt fra andre programmeringssprog. Det egner sig stort set til alle opgaver i forretningslivet og i produktionsstyring.
Programmering i FORTH er det samme som at udvide sproget med begreber, som samler mange tidligere begreber i sig selv, så et helt nyt program tilsidst kan udtrykkes med et enkelt ord. Fra starten indeholder VIC-FORTH op mod 400 ord, men det kan konstant udvides af brugeren...
Blandt fordelene ved at benytte FORTH fremhæves en lettere programudvikling.

Hjælpemiddel til den, der vil lave sine egne tegn på VIC



Tag det kvadrerede papir frem og giv dig i kast med at designe dine egne karakterer. Sådan står der som afslutning på de fleste artikler om brugerdefinerede karakterer, men VI & VIC har en bedre idé.

Lav et program, der viser dig på skærmen, hvordan de nye tegn bygges op, og lad programmet foretage beregningen af de værdier, der skal POKEs ind i karaktergeneratoren for at få de ønskede tegn frem.

Og hvis der skulle være læsere, som synes det er for besværligt, ja så er der også godt nyt til dem: VI & VIC har gjort arbejdet og tilbyder det færdige program, som denne udgaves færdigløsning til symbolsk betaling.

Brug de eksisterende

Det kan ofte være en lettelse at have noget konkret at gå ud fra, og i dette program kan man vælge at starte fra bar bund eller med de eksisterende tegn, dvs. SET 1, men ikke SET 2.
I Fl-mode skriver man blot den karakter, man gerne vil se forstørret, hvorefter programmet tegner den i stor udgave, hvor hver af de 64 felter, et tegn kan sammensættes af, kommer til at svare til den plads, ét tegn optager på skærmen.
Nedenunder skrives de data, der hører til den pågældende karakter, og i en rude ved siden af gengives tegnet i normal størrelse sammen med en oplysning om skærmkoden.

Trykker man F7, kommer man over i Definere-mode, hvor en op-pil forneden og en venstrepil til højre for det forstørrede tegn viser, hvilket felt, man netop nu kan ændre. F5 vil resultere i et sort felt, og F7 vil resultere i et hvidt felt. Bruger man F3 slettes hele tegnet - og man kan starte fra bar bund.

Mellemrumstangenten bruges til at flytte pilene uden at ændre tegnet. Når man gerne vil se resultatet af sine anstrengelser, trykker man RETURN, hvorefter programmet viser ét eksemplar af det hjemmegjorte tegn i den korrekte størrelse midt på skærmen. Ved endnu et tryk på RETURN, vender man tilbage til den forstørrede udgave samt de beregnede data for det netop viste tegn.
For at få VI & VIC's karakter-program, kan man benytte bestillingskuponen sammen med 20 kr., der vedlægges bestillingen. Man kan også sende beløbet på giro 6 17 39 93, VI & VIC, Kornager 29, 7100 Vejle.

Forskellen mellem 32 og 96 er ens. Det er kun tallet, der er forskelligt . . .

Nogle læsere, der har indtastet spillet fra VI & VIC nr. 4, er løbet ind i vanskeligheder, som vi her på redaktionen må tage vores del af skylden for.

Ingen kan nemlig se, at de mellemrum, der er imellem de grafiske karakterer i linierne 110 og 120 er skrevet med SHIFT-knappen trykket ned.
I brugervejledningen kan man i oversigten over skærmkoder se, at der er to forskellige værdier, som begge står for mellemrum (SPACE). Den eneste forskel er, at 32 er resultatet af at PRINTe mellemrummet uden at trykke på SHIFT - hvilket naturligvis er det almindeligste - mens 96 er resultatet af SHIFT SPACE.

Når der i VI & VIC's spil er anvendt 96, skyldes det, at de grafiske karakterer i linien også er SHIFT-ede, men man kunne lige så vel anvende det almindelige mellemrum.
En læser, der har indtastet spillet, fortæller, at han løste problemet ved at ændre 96 til 32 - men samme resultat ville altså kunne nås ved at bruge SHIFT-tasten i forbindelse med de nævnte linier, hvor spillebrættet tegnes på skærmen.

Pas på varmen

Det er vigtigt, at diskettestationen kan blive godt afkølet, når den er i brug.

Normalt skulle det ikke give problemer, men det er fristende at sætte den lidt af vejen - f.eks. ind i en reol, og så kan ventilationen blive for ringe. Resultatet af for dårlig ventilation bliver driftsforstyrrelser, som forsvinder, når temperaturen igen falder.

Brugerdefinerede tegn er første skridt på vejen mod højopløsnings grafik

På VIC 20 kan skærmen indstilles på to måder: normal tekst, og brugerdefinerbare tegn. Måden der anvendes er afhængig af, hvor i hukommelsen karaktergeneratoren befinder sig. Der findes også to farve-indstillinger: højopløsning og multi-farve.
De to skærmindstillinger afhænger af, hvorvidt den interne karaktergenerator anvendes, eller der benyttes en bruger defineret generator af bittene 0 til 3 i VIC-kredsens kontrolregister 5. Disse fire bit, er i virkeligheden bittene A10 til A13 i den egentlige

Bit:
3 2 1 0
Indhold:
11 01XX XXXX XXXX
Hex:
3 4 0 0

Det normale indhold af disse bit i kontrolregister 5, er nul, og sådan som VIC 20 er opbygget, svarer det til en generator adresse på Hex $8000, (decimal 32768). I denne position starter en 4K ROM, karakter generatoren, der indeholder de 256 tegns egentlige prikmønstre. 4K ROM, karakter generatoren, der indeholder de 256 tegns egentlige prikmønstre. 4K ROMmen indeholder i virkeligheden hele to generatorer, der hver fylder 2K. Den første af disse to generatorer, der starter på adresse, hex $8000 (decimal 32738), indeholder prikmønsteret for de 128 tegn. Den anden generator, der starter i hex $8800, (decimal 34816), er identisk med den første, bortset fra, at dele af de grafiske tegn, er erstattet med små bogstaver. Hvis man tilslutter den anden generator, vil VIC 20 vise små bogstaver, istedet for, som normalt, store. Store bogstaver kan her opnåes ved at trykke SHIFT tasten ned. Den anden generator aktiveres normalt fra tastaturet, ved at man nedtrykker SHIFT og COMMODORE tasterne samtidigt. Alternativt kan også anvendes følgende instruktion:
POKE 36869, 242 = sæt normal stilling, store bogst./grafik.
POKE 36869,240 = sæt anden stilling, små/store bogstaver.
Dette flytter simpelthen ge-

Her er et spil, der har brugerdefinerede figurer på skærmen

I forbindelse med VI & VIC's konkurrence om en professor-terning-simulator og diverse spil, har vi modtaget et lille spil, der arbejder med brugerdefinerede karakterer.

Der er tale om, at der kun anvendes tre karakterer, der til gengæld alle er »hjemmelavede«: En lille flyvemaskine, som styres fra side til side af cursor-kontrollerne, skal fjerne de skikkelser, som findes i øverste linie - og siden hen skal de erstattes med et monster.

Som det fremgår af programlistningen, er det Arno G. Andersen, der har sendt os spillet.

```
5 POKE36879,25:PRINT"                 ."
8 PRINT"      | .-----------. |"
10 PRINT"      | |MONSTERS| |"
13 PRINT"      | '-----------' |"
15 PRINT"      '---------------'"
20 PRINT"  MOVE LEFT  = CRSR↑"
30 PRINT"  MOVE RIGHT = CRSR←":PRINT" BY ARNO G. ANDERSEN"
40 SYS65499:WAIT162,128
50 SV=1:POKE36878,15
60 DEFFNA(X)=7680+T*22+A:SYS65499:Q=0:AN=AN+1
70 GOSUB260:PRINT"""""""""""""""""""""""""
80 FORT=38400TO38421:POKET,INT(RND(1)*6+2):NEXT
90 A=5-11*(RND(1)>.5):IFQ=22THENPRINT"":POKE36869,240:GOTO180     (192)
100 FLAG=0:FORT=21TO1STEP-SV
110 R=PEEK(197):X=A+(A>0ANDR=31)-(A<21ANDR=23):POKEFNA(1)+22*SV,252:A=X
120 POKEFNA(1),253:POKEFNA(1)+30720,2:NEXT:T=0
130 IFPEEK(FNA(1))=255THENQ=Q-1:FLAG=1
140 IFPEEK(FNA(1))<>252THENPOKEFNA(1),252:T=1:POKEFNA(1),252:GOTO240
150 POKEFNA(1)+30720,5:POKEFNA(1),255:T=1:POKEFNA(1),252:Q=Q+1
160 FORT=1TO10STEP2:FORS=225+TTO235+TSTEP5:POKE36875,S
170 FORZ=1TO2:NEXTZ,S,T:POKE36875,0:GOTO90
180 PRINT" YOU HAVE PLACED ALL      THE MONSTERS !"
190 PRINT"  TIME USED   "MID$(TI$,3,2)":"RIGHT$(TI$,2)
200 PRINT"    HIT ANY KEY":POKE198,0
210 GETA$:IFA$<>""THEN230
220 POKE38514+INT(RND(1)*14),INT(RND(1)*6+2):GOTO210
230 SV=1-(AN>=0):GOTO60
240 IFFLAGTHENFORT=250TO140STEP-2.5:POKE36876,T:NEXT:POKE36876,0:GOTO90
250 POKE36875,200:FORT=1TO99:NEXT:POKE36875,140:FORT=1TO99:NEXT:POKE36875,0:GOTO
90
260 RESTORE:FORT=7136TO7167:READA:POKET,A:NEXT:POKE36869,253:RETURN
270 DATA0,0,0,0,0,0,0,0
280 DATA16,16,16,56,124,254,16,56
290 DATA16,56,84,214,254,16,40,108
300 DATA56,124,214,214,254,254,170,170
```

neratorens startadresse 2K op i hukommelsen, hvorved den anden generator sættes i funktion.

Karaktererne ned i RAM

Karaktergeneratorens startadresse i kontrol register 5, kan ændres, således at generatoren kan lægges ned i RAM, for derved at give brugeren mulighed for at ændre tegnene, og skabe sine egne. Startadressen på brugerens egen generator, kan være starten på en hvilken somhelst 2K blok, (4K hvis der anvendes 8 × 16 tegn), der må befinde sig mellem Hex $1000 og $3000. Den bør befinde sig på den højest mulige adresse, og beskyttes mod ødelæggelse af et BASIC program, ved at sænke BASICs hukommelses pointere. Herved beskyttes det område der optages af generatoren. Når man ændrer kontrolregister 5, må følgende regler overholdes:

1 Startadressen skal altid være ved starten af en 2K blok.
2 Hvis indholdet af bit 2 og 3 er begge nul, så er startadressen automatisk sat til hex $8000, plus en forskydning, der indikeres af bit 1 og 0, denne forskydning er med 1K spring.
3 Bit 2 og 3 indeholder startadressen i spring på 4K.

Så for at sætte startadressen ved 11K, dvs. 2 × 4K blokke + 3 × 1K blokke, skal bite 0 til 3 sættes således:
Bit:
3 2 1 0
Binært indhold:
1 0 1 1
Hvilket betyder:
2 × 4K blokke
3 × 1K blokke

Den bruger definerbare karaktergenerator er meget vigtig, idet den ikke bare muliggør oprettelsen af brugerens egne tegn, men også åbner mulighed for plotning med høj opløselighed. Herved kan man tegne en kurve eller nogle liner, med en opløsning på 176 × 184 punkter, nok til at give et virkelig godt billede. Høj-opløsnings plotning opnåes ved, at anvende generator. Men brugen af den bruger-definerede, RAM-baserede generator, må først forståes fuldtud, før højopløsningsteknikken kan gennemgåes.

Fire start-adresser

Første trin i oprettelsen af egen karaktergenerator, er at finde et område i RAMen, hvor tegnene kan defineres. Hvis der er tale om 8 × 8 tegn, skal der bruges 2048 RAM-pladser, hvis det er 8 × 16, er tallet 4096. Da en standard VIC 20 kun har 3584 bytes til rådighed for brugeren, er et tegnsæt med 8 × 8 tegn, det eneste mulige. Bruger RAM-en på en uudvidet VIC starter i adresse 4096 og går til 7679. Tegngeneratoren kan bringes til at starte ved en hvilken somhelst af følgende adresser: 4096, 5120, 6144 og 7168. Da der skal bruges 2048 pladser, er den eneste mulige startadresse 5120, hvilket giver 1024 bytes tilbage, til brugeren. (Ikke meget, hvorfor det anbefales at anvende en 3K RAM-udvidelse, når man arbejder med definerede tegn. Brugen af en sådan vil ikke ændre på adresserne der nævnes her.). RAM området der nu er valgt til generatoren, skal beskyttes mod ødelæggelse af BASIC, ved at mindske værdien af pointerne til RAMmens top:

```
10 POKE 51,255 : POKE 52,19
11 POKE 55,255 : POKE 56,19
12 CLR
```

Næste trin er, at putte hver enkelt tegns data ind i det netop beskyttede område, enten med POKE instruktioner, eller med maskinkode. Men før dette kan gøres, må man sætte sig ned, og tegne hver enkelt tegn, på et kvadreret 8 × 8 ark. Se fig. 1. Når så tegnene er lavet, skal hver enkelt laves om til 8 tal, der senere skal gemmes i RAMen. Hver linie i den kvadrerede 8 × 8 tegning, skal svare til en byte, og hver enkelt bit i denne byte, skal svare til de enkelte prikker i linien. Disse tal gemmes som binære tal i hukommelsen, så ved at se på en prik som et 1-tal, og ingenting som et 0, kan de enkelte tal, bytes, altså beregnes. Se fig. 2. Det vil altså sige, at hvert enkelt tegn som vi laver, skal bestå af otte bytes. De første otte bytes, i vor generator, (der starter i position 5120), svarer til det første tegn. De næste otte bytes til det andet tegn, osv. Hvert af disse tegn får så numrene 0-255, der svarer til ASCII nummeret der skal anvendes, for at skrive tegnene på skærmen.

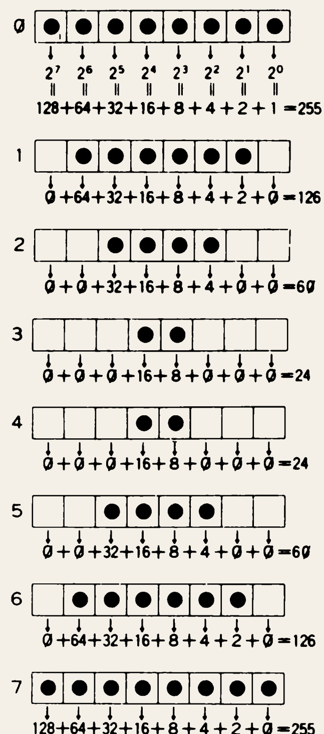
DATA-sætningen

Generatoren vi laver, behøver naturligvis kun indeholde lige netop de tegn vi har brug for, det er ikke nødvendigt at udfylde pladsen for alle 255 tegn. Men det anbefales at det første tegn starter ved begyndelsen, og at alle eventuelle »huller« udfyldes med nuller.

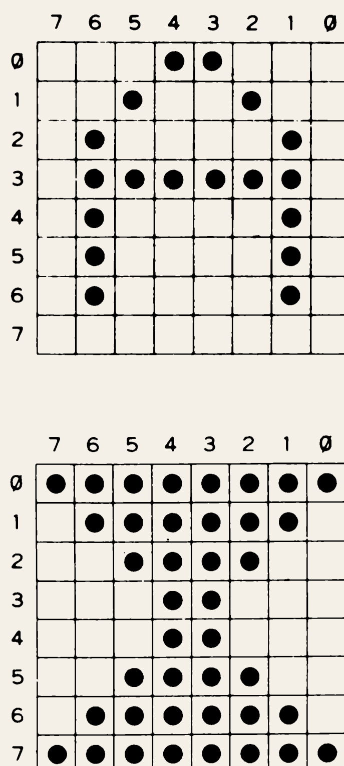
Hvis karaktergeneratoren oprettes fra et basic program, er det bedst at gemme tallene i DATA-sætninger, og så POKE dem på plads i RAMmen:

```
20 FOR I = 0 to 2048
21 READ A$
22 IF A$ = "*" THEN 30
23 POKE 5120 + I, VAL (A$)
24 NEXT
30 END
100 DATA 24, 20, 20, 28, 48,
    112, 96, 0
110 DATA 0, 24, 60, 126,
    255, 24, 36, 66
120 DATA 255, 126, 60,
    24, 24, 60, 126, 255
130 DATA *
```

I de allerfleste tilfælde har man, udover sine egne tegn, også brug for nogle af de almindelige. I sådanne tilfælde kan man hente dem direkte fra ROMmen, hvor de jo under alle omstændigheder befinder sig. De første 128 tegn



Figur 1



Figur 2

fra ROM generatoren, overføres således:

```
20 FOR I = 0 to 1024
30 POKE 5120 + I,(32768 + I)
40 NEXT
```

Herved er der mulighed for 128 bruger definerede tegn, begyndende ved 6144, der kan indlæses som beskrevet ovenfor, og tegnene vil så have en ASCII kode, der starter ved 128. Følgende er et eksempel, hvorledes man så kan indlæse karakter generatoren:

```
20 FOR I = 0 to 1024
21 POKE 5120 + I,
PEEK (32768 + I)
22 NEXT I
30 FOR I = 0 TO 1024
31 READ A$
32 IF A$ = "*" THEN 200
33 POKE 6144 + I, VAL (A$)
34 NEXT

60 REM DATA FOR
TEGNENE 128, 129 og 130
100 DATA 24, 20, 20, 28, 48,
    112, 96, 0
110 DATA 0, 24, 60, 126,
    255. 24, 36, 66
120 DATA 255, 126, 60, 24,
    24, 60, 126, 255
130 DATA *
```

Når den bruger definerede generator er indlæst i lageret, kan den bruges så længe der er spænding på VIC. Den berøres ikke af f.eks. CLR og NEW.

For at aktivere den, gives følgende instruktioner:

```
200 POKE 36869,253
210 POKE 36866, PEEK
(36866) OR 128
```

Husk farven

Det er vigtigt at bemærke sig, at hvis man bruger POKE til skærm, skal de tilsvarende positioner i farve-RAMmen også ændres, da der ellers vil skrives et hvidt tegn på hvid baggrund, det vil altså ikke kunne ses.

Prøv: POKE 7680,129:POKE 38400,4

For at komme tilbage til den normale generator, bruges følgende:

```
500 POKE 36869,240
510 POKE 36866,150
```

Dette sætter ROMmens generator i funktion, uden dog at ødelægge brugerens egen.

I næste nummer vil vi se på, hvorledes alt dette kan benyttes til at opnå høj-opløsnings grafik.

Red stumperne

Det mest bitre, der sker for en VIC programmør er næsten at måtte slukke for maskinen, vinke farvel til alle programmerne og data - bare fordi der var et eller andet, der gik galt og maskinen pludselig ikke mere var til at komme i kontakt med.

Den hyppigste årsag til den slags uheld er, at man har benyttet sig af POKE-instruktionen uden helt at have styr på, hvad der egentlig skete - eller WAIT, der kom til at vente forgæves på den ændring, der skulle starte programmet igen.

Det råd, som i hvert fald kan bruges, hvis man har båndoptager eller diskettestation er at redde stumperne. Gør det til en vane engang imellem at SAVE programmet uanset, om det er færdigt eller ej. Hvis man ganske enkelt har sin båndoptager stående med REC og PLAY knappen nedtrykket, kan man blot skrive SAVE - eventuelt efterfulgt af dagen eller tidspunktet. Hvis det skulle gå helt i skoven, vil man have et helt bånd fyldt med forskellige stadier af det program, man arbejder med...

Nye spil til VIC

Der er kommet seks nye VIC-spil på markedet. De leveres som programkapsler, der sættes direkte i VIC 20, og prisen er 285 kr. pr. spil.

De seks nye spil hedder »Adventureland«, »Pirate Cove«, »Mission Impossible«, »The Count«, »Voodoo Castle« og »Omega Race«. Til dette sidste spil kræves paddle.

Spillene er udstyret med en dansk vejledning.

Programmer på kassettebånd

Commodore Data, der forhandler VIC-computeren, er nu også begyndt at forhandle færdige programmer på kassettebånd. I den serie, der indtil nu foreligger, er der en række forskellige beregningsprogrammer, men også mere komplicerede programmer som f.eks. check-regnskab, der kan føre to konti side om side, et kartoteksprogram og et program, som kan hjælpe med at udfylde tipskuponen.

Blandt beregningsprogrammerne findes Lineær regression, CHI^2 test, Gennemsnit mm, løsning af andengrads ligning, normalfordeling, effektiv rente, nedskrivning, årlig værdiforringelse og fremtidig værdi.

Programmerne, der alle koster 75 kr., leveres med en dansk brugervejledning.

Den tænker for VIC:

Mikroprocessoren 6502 er computerens hjerne

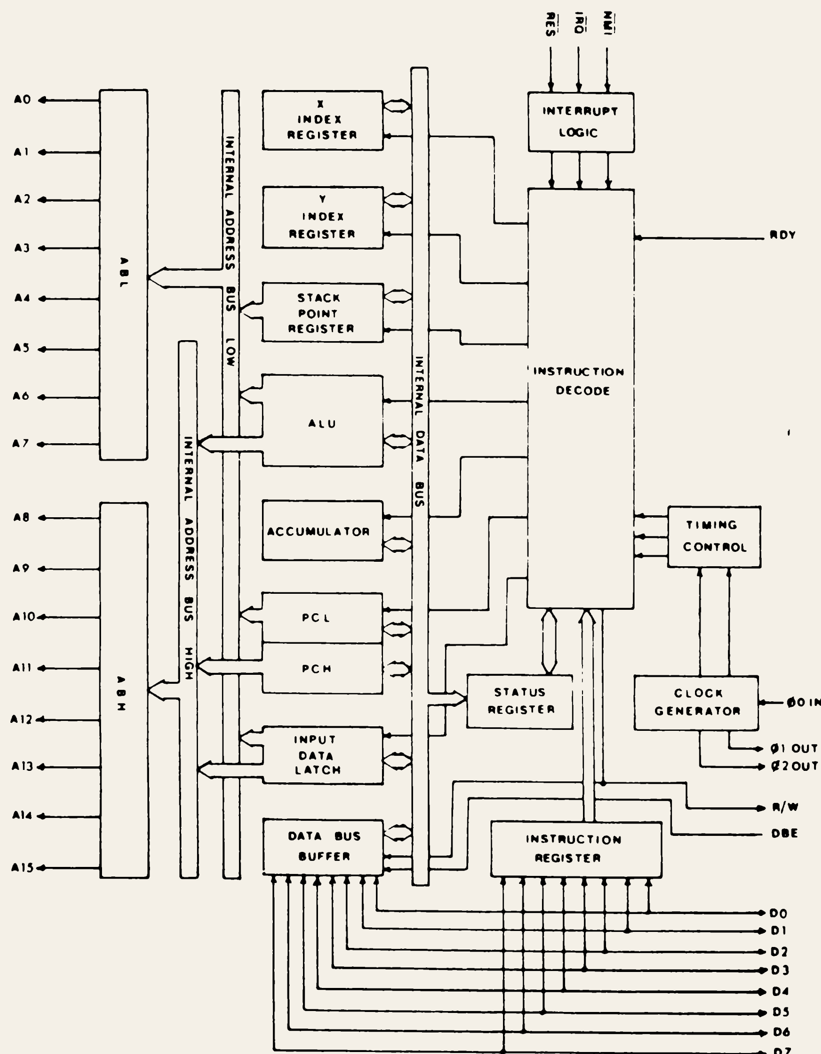
Efter vi i sidste nummer beskæftigede os med hvad maskinkode egentlig er samt det hexadecimale talsystem, vil vi i dette nummer fortsætte med en beskrivelse af det vigtigste i hele systemet, hjernen i enhver computer:
Når et program køres på VIC, udføres alle instruktionerne af een eneste komponent: mikroprocessoren. Der findes et stort udvalg af mikroprocessorer på markedet idag, men vi vil hæfte os ved een speciel, nemlig 6502, der fremstilles af MOS Technology INC. (Ikke at forveksle med MOS TEK).
MOS Technology INC, er et datterselskab af COMMODORE BUSINESS MACHINES INC., hvilket altså vil sige, at COMMODORE, som den eneste mikrocomputerproducent, råder over faciliteter, til fremstilling af egne halvlederprodukter. De såkaldte CHIP's. 6502 er en otte bit processor, hvilket vil sige, at der i hver instruktions- eller operationsperiode, arbejdes på eller overføres otte bit samtidig.

System strukturen

Fig. 1. viser et blokdiagram over den interne opbygning, (kaldet system strukturen). Det kan forekomme lidt indviklet, men det kan inddeles i to afdelinger. Den ene kaldes kontrol-afdelingen, og den anden kaldes register-afdelingen til venstre. Det er register-delen der indhenter instruktionerne fra program hukommelsen, ved hjælp af en række dataoverførsler inde i afdelingen. Hver af de 56 forskellige instruktioner, som 6502 kan genkende, har sine egne dataoverførsler, eller flytninger. Det er kontrolafdelingen der genkender instruktionerne, og iværksætter den korrekte sekvens af dataoverførsler. Instruktionerne kommer ind i processoren via databussen, hvorfra de overføres til instruktionsregistret, så de kan viderebehandles af kontrolafdelingen. Da de fleste instruktioner kræver mere end een data-overførsel indenfor register-delen, er det nødvendigt med een eller anden form for styringssignal, der sikrer af dataoverførslerne sker på rette sted og tid. Dette styres af timingdelen.
Hver eneste dataførsel, der sker i registerdelen, er et resultat af instruktions-dekoderens arbejde, og timingdelen, der tilsammen sørger for, at de rigtige registre anvendes. Når man programmerer på maskinkodeniveau, er det af største vigtighed at kunne kontrollere og manipulere data, inde i processorens registre. For at kunne forstå virkemåden af processorens instruktioner, er det vigtigt at forstå virkemåden af dens registre.

Hukommelsens udnyttelse

6502 mikroprocessoren har en 16 bit adressebus, hvilket sætter den i stand til, at bearbejde een eller 64k hukommelse. $2^{16} = 65536$. Hver eneste af de 65536 hukommelsespladser kan nåes, ved at processoren placerer det korrekte binære tal, der svarer til den ønskede plads, på adressebussen. De otte bit, der befinder sig på den adresserede plads, kan så læses, eller om nødvendigt, ændres, via da-



tabussen, der er på otte bit. Da alle processorens registre, og hukommelsen, kun er otte bit lang, er det nødvendigt med to bytes, (= 16 bit), for at angive en 16 bit hukommelsesplads. De nederste otte bit, kaldes for »high order bits«, eller mest betydende byte. Ved at dele 16 bit adressen op i to otte bit dele, kan det fulde hukommelsesområde deles op i logiske dele, eller sider. Hukommelsen indenfor hver side, kan adresseres ved at benytte den mindst betydende byte af adressen, (der jo er 16 bit lang). Heraf følger, at hver side er 256 bytes lang, og der er ialt 256 sider. (256 er det største antal man kan definere med otte bit). Side nul starter i hukommelsesplads 0, og slutter i plads 255, side et går fra 256 til 511. Bortset fra to vigtige undtagelser, er denne sideopdeling ikke vigtig for programmøren. Undtagelserne er side nul, hvortil der findes specielle instruktioner, og side et, der er processor-stak, (beskrives senere).

Det er almindeligt at udtrykke hukommelsesplaceringer, og deres indhold, i det hexadecimale talsystem. Dette er lettere end hvis man skulle angive det binært, men dog let at omskrive til binært tal, hvor det skulle være nødvendigt. Omskrivning af et decimalt tal derimod, er mere besværligt. Det er besluttet at sætte et dollar-tegn ($), foran et tal, for at angive, at det er et hexadecimalt tal, (herfra kaldet: hex-tal). I hex kan vi derfor angive enhver hukommelses-plads, eller adresse, med et fire-cifret hex-tal, hvor de to første cifre er de mest betydende, og angiver sidenummeret. Enhver dataværdi angives som et to-cifret hex-tal.

Data og programmer

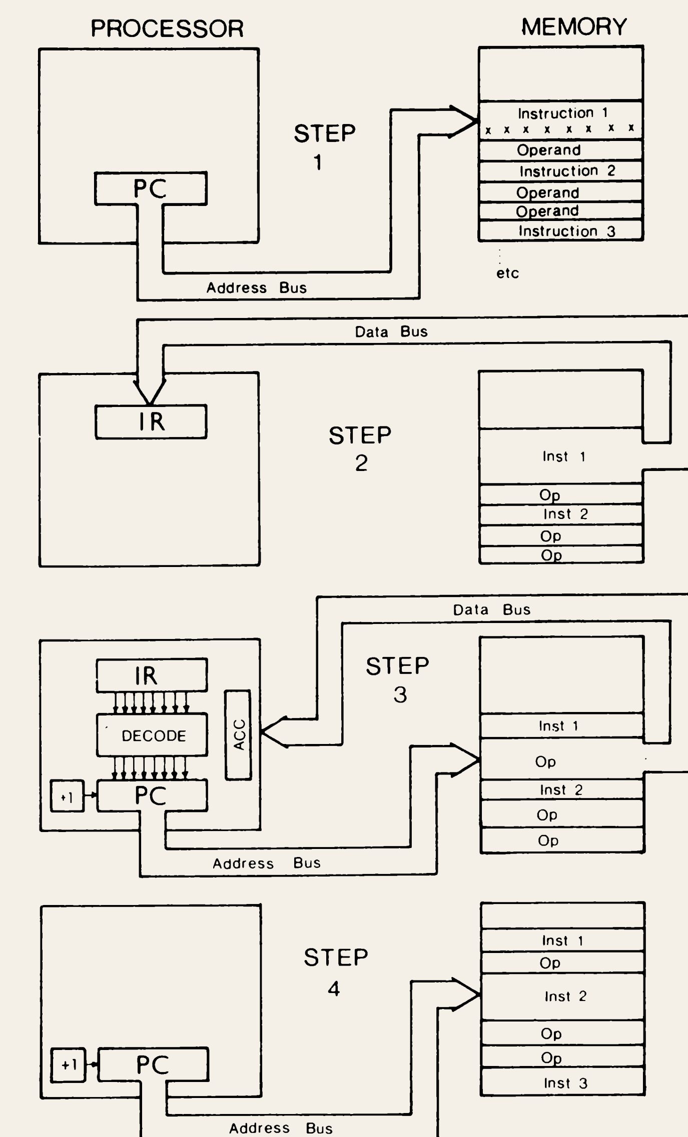
Processoren bruger hukommelsen til at gemme både data og programmer, hvor data kan være inkluderet i programmet som konstanter, eller i seperate data-tabeller. Programmer kan gemmes i enten RAM eller ROM, men data kan kun gemmes i RAM. Hver maskinkodeinstruktion i et program fylder mellem een og tre pladser. En eenbytes instruktion bearbejder altid data der er gemt i een af processorens registre. En tobyte instruktion består først af selve instruktionen, fulgt af en byte, kaldet operanden, der enten kan være en konstant, eller en side-nul adresse. En instruktion der fylder tre bytes, består først af selve instruktionen, fulgt af to bytes, der angiver en 16-bit adresse, altså en adresse indenfor processorens fulde område. Bemærk at adressen er gemt »omvendt«: den mindst betydende byte først, fulgt af den mest betydende. Når processoren skal udføre instruktionerne, der er gemt i hukommelsen, gennemgår den hver gang en fastlagt procedure. Denne procedure henter instruktionen fra hukommelsen, udfører instruktionen, og begynder så forfra, ved at hente den næste instruktion. Således fortsættes indtil slutningen på programmet er nået. Der er tre trin i denne procedure:

1) Hent instruktionen
2) Find ud af hvilken instruktion det er
3) Udfør instruktionen.

Programtælleren

Et internt register, kaldet programtælleren, er helt fundamental for denne procedure. Programtælleren indeholder altid 16-bit adressen på den næste instruktion, og første trin i proceduren er, at overføre indholdet af programtælleren til adressebussen. Instruktionen der befinder sig på denne plads, der indikeres af adressebussen, overføres til processorens instruktionsregister. Den anden fase i proceduren er, at finde ud af, hvilken instruktion der blev hentet, for at kunne lave de interne og eksterne signaler, der skal bruges. Sidste fase afhænger af instruktionen, og vil bl.a. omfatte hentning af eventuelle operander, plus manipulation af et eller flere processorregistre. Efter hentning af en instruktion eller operand, lægges een til programtælleren, således at den til sidst indeholder adressen på næste instruktion, således at proceduren kan gentages.

VIC bruger stor del af sine kræfter på at flytte data fra en plads til en anden



Akkumulatoren har ingen speciel funktion, det er et generelt mellemlager. For at flytte en data-byte fra eet sted i hukommelsen til et andet, er det nødvendigt at mellemlagre den i akkumulatoren. Akkumulatoren bruges også aritmetisk operation.

Data overførsel mellem akkumulatoren og hukommelsen, (og da VIC er såkaldt memory-mapped, gælder dette også I/O), er en meget vigtig del, og optager ialt ca. 40% af alle instruktioner, der benyttes i et maskinkodeprogram . At flytte data fra eet sted i hukommelsen til et andet kræver to instruktioner:

LDA,H1: Læs data i hukommelsesplads 1 ind i akkumulatoren.

STA,H2'; gem akkumulatorens indhold i hukommelsesplads 2.

Hukommelsespladserne H1 og H2, kan nåes på een af mange adresseringsmåder, som vi vil se på senere. Når processoren først har fået indlæst data i sig, kan processoren instrueres til at foretage aritmetiske eller logiske operationer herpå. Kun ca. 3% af et programs instruktioner er aritmetiske eller logiske.

Otte bits

Da 6502 er en otte bit processor, foregår alle aritmetiske eller logiske operationer mellem to otte bit tal. Operationerne foretages i ALU, (aritmetisk logic unit), eller regneenheden. Denne kræver, at een af operanderne er i akkumulatoren, og den anden i hukommelsen. resultatet af ALU'ens arbejde placeres i akkumulatoren. Men placeringen af resultater i en otte bit akkumulator, rejser et problem, når man f.eks. adderer to tal, hvis sum overstiger 255. Dette problem er løst ved at give akkumulatoren et niende bit, der kaldes for carry, (mente). Carry bittet, eller flag, er et bit i processorens status register, og det sættes, (= 1) når akkumulatorens indhold overskrides 255. Dette skyldes processorens udførelse af binær aritmetik, der endog som noget enestående for 6502, også kan udvides til decimal aritmetik. D.v.s. at hver byte kan indeholde to binært kodede decimale tal fra 0 til 99. Som ved binær aritmetik sættes carry-flaget, når en addition giver et resultat, der er større end 99.

Decimal-aritmetik startes med instruktionen: SED (set decimal mode), der iøvrigt sætter endnu et bit i statusregistret.

To instruktioner

Der findes to aritmetiske instruktioner:

ADC, der betyder: adder hukommelse til akkumulatoren med mente, og: SBC, der betyder: subtraher hukommelse fra akkumulator med låne. Begge instruktioner kan foretage binære eller decimale funktioner, og der kan anvendes et stort antal adresseringsmåder, for at indikere hukommelsespladsen.

ADC instruktionen adderer værdien af data i hukommelsespladsen plus værdien af carry, fra den foregående operation, til værdien i akkumulatoren. Hvis resultatet overskrider 255, eller hvis i decimal mode: 99, så sættes carry-flaget. Hvis resultatet er nul, sættes nul-flaget.

Et eksempel:

Hvis vi ønsker at addere to tal: 25 og 189, og gemme resultatet i hukommelsesplads 10, (decimal), ville vi bruge følgende instruktioner:

Instruktionerne i venstre kolonne er såkaldte mnemonics, eller forkortelser, efterfulgt af et decimalt tal, eller hukommelsesplads. Den samme sekvens vises i højre kolonne, som hex-tal. Addition af to tal, hvor een eller begge er større end 255, kræver en anden teknik, baseret på carry-flaget. Addition af to 16 bit tal, kræver to additioner. Carry slettes først, og de to mindst bety-

CLC	18	(Carry-faget slettes
LDA 25	A9 19	(Læs tallet 25 ind i akk.)
ADC 189	69 BD	(Adder 189 til akk. og carry)
STA 10	8 D 0A 00	(Gem resultatet i plads 10)

dende bytes adderes, (et 16 bit tal kræver to hukommelsespladser.). Resultatet af denne additionn gemmes i en hukommelsesplads, som den mindst betydende byte af resultatet.
dende bytes sammen med en eventuel carry opstået fra den første betydende byte sammen med en eventuel carry opstået fra den første addition, og summen gemmes som den mest betydende byte af resultatet. Med denne metode er det muligt at addere to tal af enhver størrelse, uanset om processoren regner decimalt eller binært.

Fortegn

Addition kan også foretages på to tal, eller der kan adderes to negative tal. Fortegnet gemmes som det SYVENDE bit, i den mest betydende byte, et nul for positiv fortegn, og et 1 for negativ. Additionen foregår som normalt, bortset fra, at carry for den mest betydende byte, erstattes med overflow-flaget. Dette har samme funktion, men viser et overflow eller carry fra bit syv, i stedet for bit otte.
Negative tal gemmes ikke som almindelige tal, men som 2-komplimentet af tallet. Dette beskrives bedst som det omvendte af tallet, minus 1. Alle 1'ere bliver til nul, og omvendt, undtagen bit 1. Binær 5 er normalt = 00000101, 2-komponentet er = 11111011.
SBC instruktionen subtraherer værdien af data i hukommelsen, fra værdien i akkumulatoren, med låne, (dvs. her betyder carry ikke mente, men låne).
Her bruges også 2-kompliment aritmatik. Låne-flaget, carry, er det samme som i addition, men hvor man altid sletter det før addition, sættes det altid før en subtraktion. Resultatet af en subtraktion berører carry, idet det sættes, hvis resultatet er lig med eller større end nul. Ligeledes ved subtraktion af to tal med fortegn, sættes overflow-flaget hvis resultatet overstiger +127 eller ÷127, for enkelt cifrede syv bit tal. SBC instruktionen kan anvendes til både decimal og binær subtraktion. Subtraktion af to tal, f. eks. 18 fra 27, kan foretages med følgende sekvens:

```
SED        F8          (sæt decimal mode).
SEC        38          (sæt låne flag).
LDA 27     A9 27       (Indlæs tallet 27 til akk.)
SBC 18     E9 18       (subtraher 18 fra akk. og lån).
STA 10     8D 0A 00    (gem resultat i plads 10).
```

Instruktionerne til venstre er i forkortet form, til højre som hex-tal. Bemærk at i decimal mode er hex-tallene ens.

Kun plus og minus

Instruktionssættet til 6502 indeholder ikke instruktioner, der kan dividere og multiplicere. Har man brug for disse funktioner, må man selv skrive rutiner der kan udføre dem, eller benytte de subrutiner, der allerede findes i VIC BASIC. Multiplikation foretages som gentagne additioner: 3 × 5 er det samme som: 5 + 5 + 5. For store tal kan dette blive en langsommelig proces, så der skal benyttes forskellige programmerings-tricks for at gøre det hurtigere. Division foretages som gentagne subtraktioner: 15:5 kan udføres sådan: 15-5 = 10, 10-5 = 5, 5-5 = 0. Da der skulle bruges tre subtraktioner, er resultatet altså 3. Ligesom ved multiplikation, er forskellige tricks nødvendige, for at opnå en acceptabel hastighed.

Logiske operationer

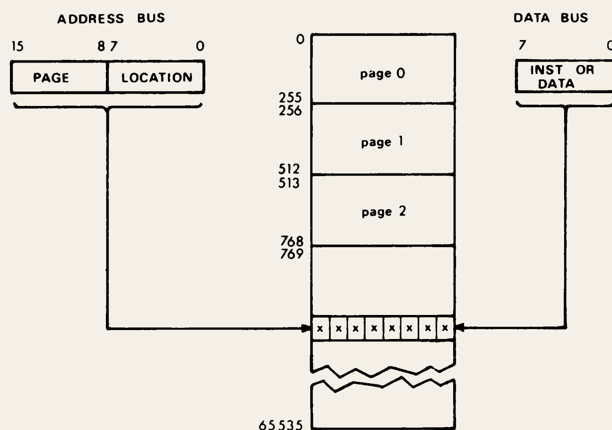
Udover aritmetiske operationer, kan ALUen også udføre logiske operationer imellem data i hukommelsen og akkumulatoren. Der er fire sådanne instruktioner: AND, BIT, OR og EOR.

AND instruktionen foretager en bit for bit AND-operation mellem en hukommelsesplads og akkumulatoren. Dvs. at bittet i akk. KUN bliver 1 hvis det i forvejen var det, SAMTIDIG med at det tilsvarende bit i hukommelsen også skal være 1. Denne instruktion kan bruges til at udtrække eet ciffer.

```
Eksempel:
LDA 25          A9 25     (indlæs tallet 25 i akk.).
AND 0F (hex)    29 0F     (AND akk. med 00001111).
STA 20          8D OA
                OO        (gem resultatet i plads 10).
```

Efter kørslen af dette program, vil plads 10 indeholde 5, idet 2-tallet er masket ud, og erstattet med 0.
BIT instruktionen er identisk med AND, bortset fra, at resultatet ikke gemmes i akkumulatoren, men kun berører status-registret.

OR instruktionen foretager en binær bit for bit OR-funktion, imellem indholdet af akkumulatoren og en hukommelsesplads. Instruktionen bruges sædvanligvis til at sætte et bit, eller grupper af bit, i en hukommelsesplads. Et logisk 1 i operanden, producerer et logisk 1 i hukommelsespladsens tilsvarende bit.
EOR, eller »exclusive OR«, er identisk med OR, bortset fra at et logisk 1 i operationen, giver et logisk 0. Og omvendt. EOR benyttes derfor ofte til at producere et tals 2-komponent.



Instruction	Memory Contents	Function	Start Address
LDA#$02	$A9	INST	+ 1
	$02	OP	+ 2
STA $0253	$8D	INST	+ 3
	$53	OP LSB	+ 4
	$02	BP MSB	+ 5
CLC	$18	INST	+ 6
ADC #$50	$69	INST	+ 7
	$50	OP	+ 8
STA z$55	$85	!NST	+ 9
	$55	OP	+ 10
RTS	$60	INST	+ 11

VIC programmer

Lad andre få mulighed for at købe de programmer, du udvikler til VIC.

Rubrikannoncerne her i bladet koster 1 kr. pr. spaltemillimeter, og for overskuelighedens skyld bør annoncerne bygges op på samme måde, som det kan ses her på siden. Det vil sige, at annonceteksten skal indeholde oplysninger om nødvendigt udstyr, hvilket medium, programmet findes på (kassettebånd, disk eller cartridge), prisen - og naturligvis oplysninger om, hvor man kan købe programmet.

Brug kuponen til din annonce.

Moonshuttle

Försök landa månlandaren utan att träffa meteorerna. Välj svårighet själv.
Hardware: VIC-20 (3583 bytes).
Medium: Kassettebånd.
Pris: Dk 35:- forudbetalt.
Købes hos: Tore Gullstrand, Långsjöv 59, S-125 30 Älvsjö, Sweden.

Mini - Pacman

Sammla poäng och undvik Spöket (inom 2 minuter).
Hardware: VIC-20 3583k.
Medium: Kassettebånd.
Pris: Dk 50:- forudbetalt.
Købes hos: Tore Gullstrand. Långsjöv 59, S-125 30 Älvsjö, Sweden.

Asteroids

Kom helskindet gennem meteorstromen og red astronauterne fra den sikre død. To hastigheder, highscore, selvdefineret grafik og gode lydeffekter.
Hardware: VIC-20 (3583 bytes).
Medium: Kassettebånd.
Pris: 75 kr. (check vedlagt bestilling).
Købes hos: VICSOFT, Krumningen 11, 2950 Vedbæk, telf. (02) 89 19 83.

TIPS

Udskriver efter % fordeling tipsrk. enkeltvis eller som hele kuponer. finder selv antal rigtige når 13' tastes ind. 3k RAM (evt. Superexp) nødvendig. Vejledning medfølger.
Hardware: VIC 20 + 3k RAM.
Medium: Cassette.
Pris: 85,- frit leveret ved forudbetaling.
Købes hos: O. Böttcher, Byløkken 34, 8240 Risskov.

Rocks

Skjut ner meteorerna och fienden innan dom träffar dig.
Hardware: VIC-20 (4000 bytes).
Medium: Kassettebånd.
Pris: Dk 50:- forudbetalt.
Købes hos: Tore Gullstrand, Långsjöv 59, S-125 30 Älvsjö, Sweden.

SIMON

Simon er et udfordrende spil, som appellerer til din hukommelsesevne. Spillet går ud på, at man skal huske en rækkefølge af toner og farver, som computeren viser. Efterhånden som spillet udvikler sig bliver der flere toner at huske, og hastigheden øges også.
Hardware: VIC 20 (Udvidet).
Medium: Kassettebånd.
Pris: Kr. 75,- ved forudbetaling. Ved køb pr. efterkrav tillægges postvæsenets takster.
Købes hos: Jens Mikkelsen, Adelgade 26, 8660 Skanderborg.

VIC program sælges

Programmets navn: ____________________

Beskrivelse: ____________________

____

____

____

____

____

____

____

Hardware: ____________ **Medium:** ____________

Pris: ____________________

Købes hos: ____________________

____

Kuponen sendes til VI & VIC, Kornager29, 7100 Vejle.

VIC-20

Hjemmecomputeren der vokser med opgaverne



1 VIC-20 computer
Det private computersystem starter med VIC-20 - hjemmecomputeren fra Commodore. Verdens første fuldt udbyggede farvecomputer til **kr. 2.995,-**

VIC-20 kan udbygges til et enkelt eller mere avanceret system efter ønske og i takt med brugerens behov. Til privat brug. Til undervisning. Til forretningsbrug.

VIC-20 har skrivemaskinetastatur. 8 funktionstaster. 255 farvekombinationer. 64 grafiske tegn. Musikgenerator. Intern hukommelse op til 32K RAM. Tilsluttes alle TV-apparater og monitorer.

2 VIC-1530 kassettestation
Det første skridt mange tager, når de udbygger deres VIC-computersystem. Kassettestationen kan bruges til datalager eller til overførsel af programmer. **Pris: kr. 1.035,-.**

3 VIC-1540 diskettestation
Diskettestationen giver mulighed for det fulde udbytte af VIC-computersystemet. Lagerkapaciteten udvides til 170K bytes. Op til 4 diskettestationer kan kobles til samtidigt. Velegnet til programudvikling. Hurtig dataoverførsel. **Pris: kr. 6.495,-.**

4 VIC-1515 printer
Rigtig papirprinter. 80 karakterer og en hastighed på 30 tegn/sek. **Pris: kr. 4.945,-.**

5 VIC-1020 moder-modul
Dette modul er bl.a. nødvendigt for at udnytte VIC fuldt ud. Gør det muligt samtidig at tilslutte f.eks. 3 - 8 og 16K RAM, superexpander, etc. **Pris: kr. 2.280,-.**

6 Udvidelse af hukommelse
VIC-20's hukommelseskapacitet kan udvides med 3K til 27K RAM.

7 VICMON
Gør det muligt at programmere i maskinsprog. **Pris: kr. 400,-.**

8 VIC-1211A Superexpander
Giver højopløst grafik og ekstra kommandoer til plotning, farvelægning, etc. **Pris: kr. 595,-.**

9 VIC-1212 Programmers aid pack
Et godt hjælpeværktøj ved programmering. Giver 14 ekstra kommandoer. **Pris: kr. 495,-.**

10 VIC-1311 joy stick
Til spil og lignende. **Pris: kr. 95,-.**

11 Brugerhåndbog
Der findes let forståelig dansk brugerhåndbog til VIC, som også på enkel og klar måde fortæller om BASIC. Hveranden måned udkommer brugerbladet VI & VIC, som kan købes i løssalg eller abonnement med VIC-nyheder.

Find nærmeste autoriserede VIC-forhandler på listen og kom ind og prøv VIC hjemmecomputeren.

Alle priser er incl. moms.

commodore COMPUTER

05 - 64 11 55 eller 01 - 88 15 05

ØERNE: Ballerup: Mic Kasseapparater, 02-655111. **Bornholm:** Krogh Hansens Radio & TV, 03-998127. **Herlev:** W. Rolf Pedersen, 02-915511. **Hillerød:** Bo-el Data, 02-253131. **Holbæk:** Papirgården, 03-433003. **København:** BN Elektronic, 01-811900 òg 01-184555. CR-Commander Radio, 01-343422. L. A. Elektronik Kbh., 01-551540. Betafon Radio, 01-310273. Samfundslitteratur, 01-351942. Poly Data Microcenter, 01-420705. W. Rolf Pedersen, 01-137056. **Lyngby:** BN Elektronik, 02-881900. Polyteknisk Boghandel, 02-881488. **Maribo:** JD Totalinformation, 03-881700. **Odense:** Dansk Totalinformation, 09-126488. EDB Butikken, 09-147660. Fl. Kjerulff, 09-135480. **Otterup:** Werners Radio Elektronik, 09-823333. **Ringsted:** Ahrent Flensborg, 03-610011. **Rødovre:** Datacare, 01-705858. **Slagelse:** Vest Sjællands Mikrodata, 03-534707. **St. Heddinge:** JE Kontorcenter, 03-703332. **Svendborg:** Jens Bach Kontordata, 09-212788. **Viby Sj.:** Baagøe Radio, 02-394100. **Vordingborg:** Kontorforsyningen, 03-770170. **GRØNLAND: Godthåb:** Godthåb Elektronikservice, 21865. **Julianehåb:** Musikhuset Julianehåb, 01-438133. **JYLLAND: Aalborg:** Knud Engsig Kontorudstyr, 08-126666. H. C. Elektronic, 08-142314. **Esbjerg:** Kurt Øland, 05-125299. **Fredericia:** Semicap Data, 05-931565. **Gram:** Teknik Huset, 04-822711. **Grenå:** Djursland Data, 06-326465. **Haderslev:** H. C. Reimers, 04-523714. **Hadsten:** Hadsten Elektronik, 06-980408. **Herning:** Logic Design, 07-221300. **Hjørring:** Norad, 08-960188. **Holstebro:** Pe-Co Computer System, 07-414646. **Horsens:** Dansk Computer Center, 05-611110. **Kolding:** Hauge Rasmussen Foto, 05-520070. **Randers:** Djursland Data, 06-434200. **Ringkøbing:** LP Musik, 07-322511. **Skanderborg:** Bildata, 06-525344. **Skive:** Elektroniklageret, 07-526177. **Struer:** Helmholt Elektronik, 07-852611. **Sønderborg:** G. P. Kontorsystemer, 04-424546. **Vejle:** 2R Data, 05-831600. **Århus:** Akademisk Boghandel, 06-128844. Clemens Papirteknik, 06-133922. Editor, 06-127720.

Hedelund Marketing, Vejle